

# DEVICE AND METHOD FOR CONTROLLING BOOT OF CPU

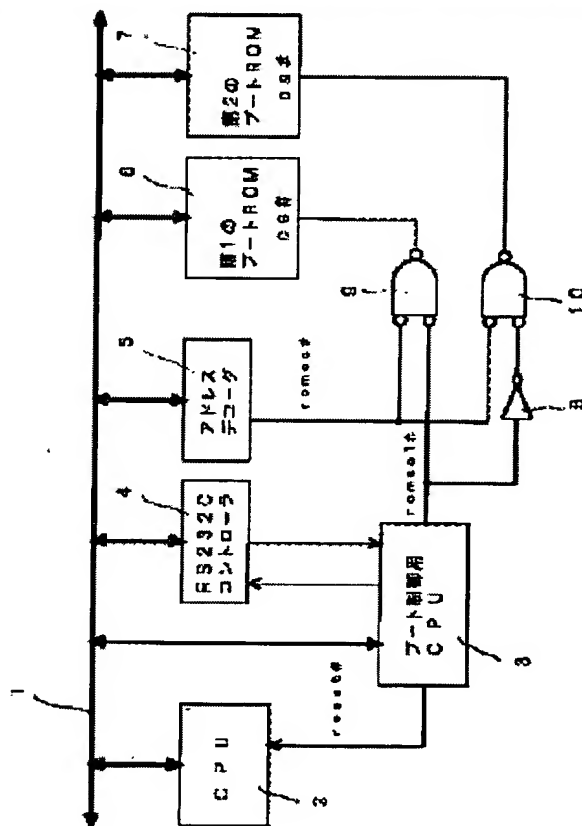
**Patent number:** JP2001109629  
**Publication date:** 2001-04-20  
**Inventor:** SAKAMOTO NAOMASA  
**Applicant:** TOSHIBA CORP  
**Classification:**  
 - International: G06F9/445; G06F12/16  
 - european:  
**Application number:** JP19990283662 19991005  
**Priority number(s):**

Report a data error here

## Abstract of JP2001109629

**PROBLEM TO BE SOLVED:** To provide a device and a method for controlling boot of CPU with which a CPU can be booted even when a memory storing a boot program is destroyed.

**SOLUTION:** Concerning the boot controller for CPU, electronic equipment having a CPU is provided with a first boot ROM storing the boot program of the CPU, second boot ROM storing the boot program of the CPU, first reading means for reading the boot program out of the first boot ROM when a power source is turned on, judging means for judging whether the CPU can be normally booted or not from the boot program read out of the first boot ROM by the first reading means, and means for reading the boot program out of the second boot ROM based on the result of this judging means when the CPU can not be booted.



Data supplied from the esp@cenet database - Patent Abstracts of Japan

Best Available Copy

(18) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2001-109629

(P2001-109629A)

(43) 公開日 平成13年4月20日(2001.4.20)

(51) Int. Cl. <sup>7</sup>	識別記号	F I	テ-マ-ト (参考)
G 0 6 F 9/445		G 0 6 F 12/16	3 1 0 J 5 B 0 1 9
12/16	3 1 0	9/06	4 2 0 S 5 B 0 7 6

審査請求 未請求 請求項の数4 O L (全 7 頁)

(21) 出願番号 特願平11-283882

(22) 出願日 平成11年10月5日(1999.10.5)

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72) 発明者 坂本 直正

東京都青森市木広町2丁目9番地 株式会社

東芝青森工場内

(74) 代理人 100083161

弁護士 外川 英明

Fターム(参考) 5B018 GA04 HA04 KA13 MA23 NA06

QA11

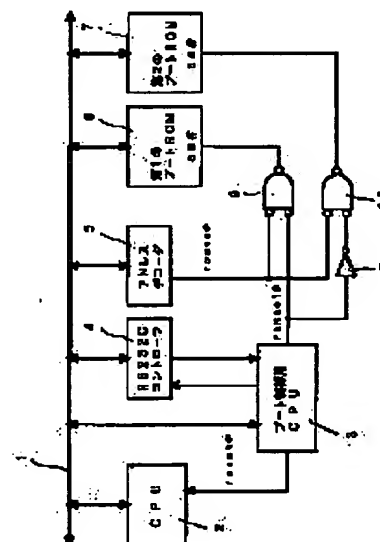
5B078 AA01 AB17 BB17 CA01 CA06

(54) 【発明の名称】 CPUのブート制御装置及びブート制御方法

(57) 【要約】

【課題】 本発明は、ブートプログラムを記憶したメモリが壊れた場合でも、CPUをブートすることができるCPUのブート制御装置及びブート制御方法を提供することを目的とする。

【解決手段】 CPUを持つ電子機器において、前記CPUのブートプログラムが記憶された第1のブートROMと、前記CPUのブートプログラムが記憶された第2のブートROMと、電源がONされた際に、前記第1のブートROMからブートプログラムを読み出す第1の読み出し手段と、前記CPUが前記第1の読み出し手段により前記第1のブートROMから読み出したブートプログラムにより正常にブートができたか否かを判断する判断手段と、この判断手段の結果に基づいて、前記CPUがブートできなかった場合に、前記第2のブートROMからブートプログラムを読み出す手段とを設けたCPUのブート制御装置。



【特許請求の範囲】

【請求項1】 CPUを持つ電子機器において、  
前記CPUのブートプログラムが記憶された第1のブートROMと、  
前記CPUのブートプログラムが記憶された第2のブートROMと、  
電源がONされた際に、前記第1のブートROMからブートプログラムを読み出す第1の読み出し手段と、  
前記CPUが前記第1の読み出し手段により前記第1のブートROMから読み出したブートプログラムにより正常にブートができたか否かを判断する判断手段と、  
この判断手段の結果に基づいて、前記CPUがブートできなかった場合に、前記第2のブートROMからブートプログラムを読み出す手段とを具備したことを特徴とするCPUのブート制御装置。

【請求項2】 CPUを持つ電子機器において、  
前記CPUのブートプログラムがそれぞれに記憶された第1及び第2の領域を持つブートROMと、  
電源がONされた際に、前記ブートROMの前記第1の領域からブートプログラムを読み出す第1の読み出し手段と、  
前記CPUが前記第1の読み出し手段により前記ブートROMの前記第1の領域から読み出したブートプログラムにより正常にブートができたか否かを判断する判断手段と、  
この判断手段の結果に基づいて、前記CPUがブートできなかった場合に、前記ブートROMの前記第2の領域からブートプログラムを読み出す第2の読み出し手段とを具備したことを特徴とするCPUのブート制御装置。

【請求項3】 CPUを持つ電子機器において、  
前記CPUのブートプログラムがそれぞれ記憶された第1及び第2のブートROMを設け、  
電源がONされた際に、前記第1のブートROMからブートプログラムを読み出し、  
この読み出したブートプログラムにより前記CPUが正常にブートができたか否かを判断し、  
この判断手段の結果に基づいて、前記CPUがブートできなかった場合に、前記第2のブートROMからブートプログラムを読み出すことを特徴とするCPUのブート制御方法。

【請求項4】 CPUを持つ電子機器において、  
前記CPUのブートプログラムがそれぞれに記憶された第1及び第2の領域を持つブートROMを設け、  
電源がONされた際に、前記ブートROMの前記第1の領域からブートプログラムを読み出し、この読み出したブートプログラムにより前記CPUが正常にブートができたか否かを判断し、  
この判断手段の結果に基づいて、前記CPUがブートできなかった場合に、前記ブートROMの前記第2の領域からブートプログラムを読み出すことを特徴とするCPUのブート制御方法。

Uのブート制御方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、パーソナルコンピュータを初めとする電子機器の制御の中核として使用されているマイクロプロセッサ（以下、CPUと呼ぶ）のブート制御の改良に関する。

【0002】

【従来の技術】 近年パーソナルコンピュータ等においてCPUのブートプログラムを記憶するリードオンリメモリとしてフラッシュメモリを使用してブートプログラムの書き換えをすることができるように構成しているシステムがある。

【0003】 しかし、このようなシステムでは、ブートプログラムの書き換えに何らかの原因で失敗すると、フラッシュメモリが壊れてしまい、CPUが正しくブートできなくなり、パーソナルコンピュータ等のシステム（電子機器）が立ち上がらなくなってしまうという問題が生じていた。このため従来では、フラッシュメモリを交換して新たにブートプログラムを記憶させて、再立ち上げていた。

【0004】

【発明が解決しようとする課題】 本発明は、このような問題を解決するためになされたもので、ブートプログラムを記憶したメモリが壊れた場合でも、CPUをブートすることができるCPUのブート制御装置及びブート制御方法を提供することを目的とする。

【0005】

【課題を解決するための手段】 本発明は、CPUを持つ電子機器において、前記CPUのブートプログラムが記憶された第1のブートROMと、前記CPUのブートプログラムが記憶された第2のブートROMと、電源がONされた際に、前記第1のブートROMからブートプログラムを読み出す第1の読み出し手段と、前記CPUが前記第1の読み出し手段により前記第1のブートROMから読み出したブートプログラムにより正常にブートができたか否かを判断する判断手段と、この判断手段の結果に基づいて、前記CPUがブートできなかった場合に、前記第2のブートROMからブートプログラムを読み出す手段とを設けたことを特徴とする。

【0006】 このように本発明によれば、CPUが確実にブートすることができる。本発明は、CPUを持つ電子機器において、前記CPUのブートプログラムがそれぞれに記憶された第1及び第2の領域を持つブートROMと、電源がONされた際に、前記ブートROMの前記第1の領域からブートプログラムを読み出す第1の読み出し手段と、前記CPUが前記第1の読み出し手段により前記ブートROMの前記第1の領域から読み出したブートプログラムにより正常にブートができたか否かを判断する判断手段と、この判断手段の結果に基づいて、前

記CPUがブートできなかった場合に、前記ブートROMの前記第2の領域からブートプログラムを読み出す第2の読み出し手段とを設けたことを特徴とする。このように本発明によれば、CPUが確実にブートすることができる。

【0007】

【発明の実施の形態】以下、図面を参照して本発明の実施形態について説明する。図1は、本発明の第1の実施形態を示すシステム図である。図1において、システムバス1にはそれぞれシステム全体の制御を行うCPU2、CPU2のブート制御用CPU3、RS232Cコントローラ4、アドレスデコーダ5、第1のブートROM6、第2のブートROM7とが接続されている。ブート制御用CPU3には、CPU1のブート制御を行うプログラムを予め内部に記憶している。このプログラムの動作については、後程詳述する。RS232Cコントローラ4は、CPU1から出力される通信用のデータをシステムバス1を介して受信し、この受信した通信データをシリアルデータとしてブート制御用CPU3に出力する。第1のブートROM6及び第2のブートROM7は、フラッシュメモリで構成されたCPU2のブートプログラムを記憶するものであり、この2つのメモリには、同一内容のブートプログラムを記憶しておく。通常CPU2がそのブートシーケンスを実行する場合には、第1のブートROM6からブートプログラムを読み出すよう構成されている。アドレスデコーダ5は、図示しない電源がONになり、図示しないパワーオンリセット回路が動作してCPU2がパワーオンリセットされ、CPU2が第1のブートROM6からブートプログラムを読み出すために出力したアドレス信号をデコードして第1のブートROM6を選択するためのチップセレクト信号(romcs#)を出力するものである。このチップセレクト信号の信号名romcs#の最後に「#」が付与されているのは、このチップセレクト信号romcs#がローアクティブの信号であることを示している。以後信号名の最後に「#」が付与されているものは、ローアクティブの信号である。このアドレスデコーダ5から出力されたチップセレクト信号romcs#は、第1のANDゲート回路9及び第2のANDゲート回路10の第1の入力端子に入力されている。ブート制御用CPU3からは、ROM選択信号romsel#を出力する。このROM選択信号romsel#は、第1のANDゲート回路9の第2の入力端子に入力されているとともに、NOT回路8を介して第2のANDゲート回路10の第2の入力端子に入力されている。第1のANDゲート回路9の出力信号は、第1のブートROM6のチップセレクト信号cs#として第1のブートROM6に供給されている。また、第2のANDゲート回路10の出力信号は、第2のブートROM7のチップセレクト信号cs#として第2のブートROM7に供給されている。更にブ

ート制御用CPU3からCPU2に対してリセット信号reset#が供給されるよう接続されている。

【0008】このように構成されたシステムにおけるCPU2のブート動作について、図2に示したフローチャート図を用いて詳細に説明する。まず、システムの図示しない電源がONされて図示しないパワーオンリセット回路が動作してCPU2がパワーオンリセットされ、CPU2が第1のブートROM6からブートプログラムを読み出すための動作を開始する(ステップS20)。

【0009】そこでCPU2はブートプログラムを第1のブートROM6から読み出すために、システムバスにアドレスを出力して第1のブートROM6の読み出し動作を開始する。この読み出し動作の開始に伴って、アドレスデコーダ5から第1のブートROM6を選択するためのチップセレクト信号(romcs#)がローレベルで出力される。この時点でブート制御用CPU3から出力されているROM選択信号romsel#の信号のレベルはローであるため、第1のANDゲート回路9から第1のブートROM6にローレベルのチップセレクト信号cs#が出力されて第1のブートROM6が選択状態になる。この選択によって、CPU2は第1のブートROM6からブートプログラムを読み出して、ブートシーケンスの実行を開始する。このとき、第2のANDゲート回路10からは、ハイレベルのチップセレクト信号cs#が出力されるため、第2のブートROM7は選択状態にならない。一方、ブート制御用CPU3は、内蔵するカウンタタイマの動作を開始させてCPU2のブートシーケンスに費やしている時間の監視を始める(ステップS21)。

【0010】次にステップS22において、上記時間監視の結果、カウンタタイマの動作が開始されてから所定の規定時間内にCPU2からRS232Cコントローラ4を介して通信データが送信されてくるか監視する(ステップS22)。これはCPU2が正常にブートシーケンスを実行して正常に立ち上がった場合に、CPU2はブート制御用CPU3に対してRS232Cコントローラ4を介して所定の通信データを送信することでCPU2が正しくブートしたことの状態を知らせてくる仕組みである。この仕組みは予めブートシーケンスの一部にその処理ルーチンを組み込むことで実現する。

【0011】このステップS22の判定の結果、CPU2が規定時間内に正常にブートシーケンスを実行して立ち上がった場合には、ブート制御用CPU3の動作は終了する。一方、第1のブートROM6に記憶しているブートプログラムがアップデート等のために書き換えられた直後に電源がONされたときに、その書き換えが何らかの原因で失敗した場合等の理由からCPU2が規定時間内に正常にブートシーケンスを実行して立ち上がらなかった場合には、ステップS23に進む。

【0012】ステップS23において、ブート制御用C

PU 3は、CPU 2に対してローレベルのリセット信号  $\text{reset}\#$  を出力して、CPU 2をリセットする（ステップS23）。次にフート制御用CPU 3は、その出力するROM選択信号  $\text{romsel}\#$  のレベルをローレベルからハイレベルに切り替える（ステップS24）。続いてフート制御用CPU 3は、その出力しているリセット信号  $\text{reset}\#$  をハイレベルにしてCPU 2のリセット状態を解除する（ステップS25）。このようにリセット状態を解除されるとCPU 2は再びパワーオンリセット状態になり、再度第1のフートROM 6からフートプログラムを読み出すための動作を開始する。従って、CPU 2はフートプログラムを第1のフートROM 6から読み出すために、システムバスにアドレスを出力して第1のフートROM 6の読み出し動作を開始する。この読み出し動作の開始に伴って、アドレスデコーダ5から第1のフートROM 6を選択するためのチップセレクト信号（ $\text{romos}\#$ ）がローレベルで出力される。この時点ではフート制御用CPU 3から出力されているROM選択信号  $\text{romsel}\#$  の信号がハイレベルであるため、第2のANDゲート回路10から第2のフートROM 7にローレベルのチップセレクト信号  $\text{os}\#$  が出力されて第2のフートROM 7が選択状態になる。この選択によって、CPU 2は第2のフートROM 7からフートプログラムを読み出して、フートシーケンスの実行を開始する。このとき、第1のANDゲート回路9からは、ハイレベルのチップセレクト信号  $\text{os}\#$  が出力されるため、第1のフートROM 6は選択状態にならない。このようにCPU 2が第2のフートROM 7からフートプログラムを読み出すことにより、フートシーケンスを実行して正常に立ち上げをすることができるようになる。

【00.13】次に、第2の実施形態について図3を用いて詳細に説明する。図3において、図1と同一構成には、同一参照符号を付してあり、詳細な説明は省略する。一方、図1に示した第1の実施形態との違いについて説明する。フートROM 11は、例えば128KB×8bit構成のフラッシュメモリで構成されている。そして、このフートROM 11のメモリ領域の下位領域の64KB（アドレス領域00.0000.h～0FFFF.h）と上位領域の64KB（アドレス領域10000.h～1FFFF.h）に分割し、この分割した上位領域と下位領域の双方に64KBからなる同一のフートプログラムを記憶させておく。この2つの領域のどちらに記憶されたフートプログラムを読み出すかは、入力するアドレス信号のうちのアドレス信号A16の1ビットの信号が、ローレベルかハイレベルかによって切り替えられる。通常、フートROM 11の下位領域に記憶されているフートプログラムを読み出して使用するために、アドレス信号A16はローレベルとされている。このフートROM 11のアドレス信号A16の入力端子には、バッファゲート

ート回路12とバッファゲート回路13の出力がワイヤードORされて入力されている。

【00.14】バッファゲート回路12とバッファゲート回路13の入力端子には、システムバス1からフートROM 11に与えられるアドレス信号のうちのアドレス信号A16の1ビットが入力されている。フート制御用CPU 3から出力されているROM選択信号  $\text{romsel}\#$  は、NOTゲート回路14とバッファゲート回路12の制御信号入力端子Gにそれぞれ入力されている。また、NOTゲート回路14の出力信号は、バッファゲート回路13の制御信号入力端子Gに入力されている。バッファゲート回路12は、制御信号入力端子Gに入力している制御信号がローレベルのときには、入力した信号をその出力端子から出力する。一方、制御信号入力端子Gに入力している制御信号がハイレベルのときには、その出力端子はハイインピーダンス状態になる。バッファゲート回路13の制御信号入力端子Gに入力している制御信号がローレベルのときには、入力した信号のレベルを反転させてその出力端子から出力する。一方、制御信号入力端子Gに入力している制御信号がハイレベルのときには、その出力端子はハイインピーダンス状態になる。尚、フートROM 11はフラッシュメモリで構成されているため、上位領域と下位領域はそれぞれ別々にその記憶内容を書き換えることが可能となっている。

【00.15】このように構成されたシステムにおけるCPU 2のフート動作について、図2に示したフローチャート図を用いて詳細に説明する。まず、システムの図示しない電源がONされて図示しないパワーオンリセット回路が動作してCPU 2がパワーオンリセットされ、CPU 2がフートROM 11からフートプログラムを読み出すための動作を開始する（ステップS20）。CPU 2はフートROM 11のアドレス00.0000.hからフートプログラムを読み出すものとする。

【00.16】そこでCPU 2はフートプログラムをフートROM 11から読み出すために、システムバスにアドレスを出力してフートROM 11からの読み出し動作を開始する。この読み出し動作の開始に伴って、システムバス1からフートROM 11に与えられるアドレス信号A16がローレベルで出力され、バッファゲート回路12とバッファゲート回路13の入力端子に入力されている。この時点でバッファゲート回路12の制御入力端子Gには、フート制御用CPU 3から出力されているローレベルのROM選択信号  $\text{romsel}\#$  が供給されているため、バッファゲート回路12の出力端子からその入力したローレベルのアドレス信号A16がフートROM 11のアドレス信号端子A16に与えられる。このため、CPU 2は、フートROM 11の下位領域からフートプログラムを読み出して、フートシーケンスの実行を開始する。一方、この時点でバッファゲート13の制御入力端子Gには、ローレベルのROM選択信号  $\text{roms}$

e1#がANDゲート回路14を介して反転されて入力されているので、バッファゲート13の出力端子はハイインピーダンス状態になっている。一方、ブート制御用CPU3は、内蔵するカウンタタイマの動作を開始させてCPU2のブートシーケンスに費やしている時間の監視を始める（ステップS21）。

【0017】次にステップS22において、上記時間監視の結果、カウンタタイマの動作が開始されてから所定の規定時間内にCPU2からRS232Cコントローラ4を介して通信データが送信されてくるか監視する（ステップS22）。このステップS22の判定の結果、CPU2が規定時間内に正常にブートシーケンスを実行して立ち上がった場合には、ブート制御用CPU3の動作は終了する。一方、ブートROM11に記憶しているブートプログラムがアップデート等のために書き換えられた直後に電源がONされたときに、その書き換えが何らかの原因で失敗した場合等の理由からCPU2が規定時間内に正常にブートシーケンスを実行して立ち上がらなかった場合には、ステップS23に進む。

【0018】ステップS23において、ブート制御用CPU3は、CPU2に対してローレベルのリセット信号reset#を出力して、CPU2をリセットする（ステップS23）。次にブート制御用CPU3は、その出力するROM選択信号romsel#のレベルをローレベルからハイレベルに切り替える（ステップS24）。続いてブート制御用CPU3は、その出力しているリセット信号reset#をハイレベルにしてCPU2のリセット状態を解除する（ステップS25）。このようにリセット状態を解除されるとCPU2は再びパワーオンリセット状態になり、再度ブートROM11からブートプログラムを読み出すための動作を開始する。

【0019】従って、CPU2はブートプログラムをブートROM11から読み出すために、システムバス1にアドレスを出がしてブートROM11の読み出し動作を開始する。この読み出し動作の開始に伴って、システムバス1からブートROM11に与えられるアドレス信号

A16がローレベルで出力され、バッファゲート回路12とバッファゲート回路13の入力端子に入力される。この時点でブート制御用CPU3から出力されているROM選択信号romsel#の信号レベルがハイレベルであるため、バッファゲート回路12の出力端子はハイインピーダンス状態になっている。一方、バッファゲート回路13の出力信号には入力したアドレス信号A16が反転され、ハイレベルのアドレス信号A16が出力される。これによって、ブートROM11のアドレス信号端子A16には、ハイレベルのアドレス信号A16が与えられる。このため、CPU2は、ブートROM11の上位領域からブートプログラムを読み出して、ブートシーケンスの実行を開始する。

【0020】このようにCPU2がブートROM11の上位領域からブートプログラムを読み出すことにより、ブートシーケンスを実行して正常に立ち上げをすることができるようになる。

【0021】

【発明の効果】以上説明した通り、本発明によればブートプログラムの書き換えに伴ってメモリが壊れた場合でも、CPUをブートすることができる。

【図面の簡単な説明】

【図1】本発明の第1の実施形態に関わるシステム構成を示す図である。

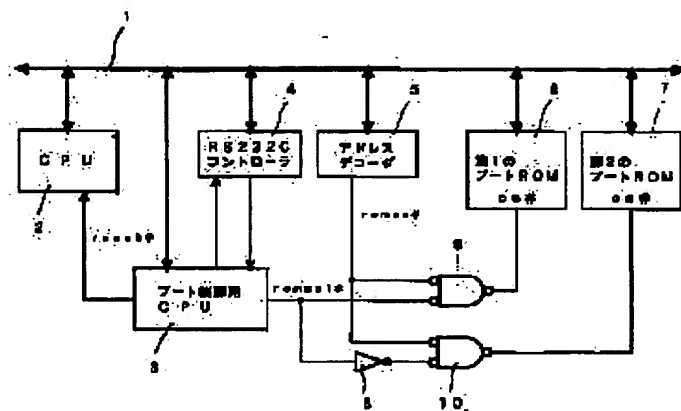
【図2】本発明の動作を説明するためのフローチャート図である。

【図3】本発明の第2の実施形態に関わるシステム構成を示す図である。

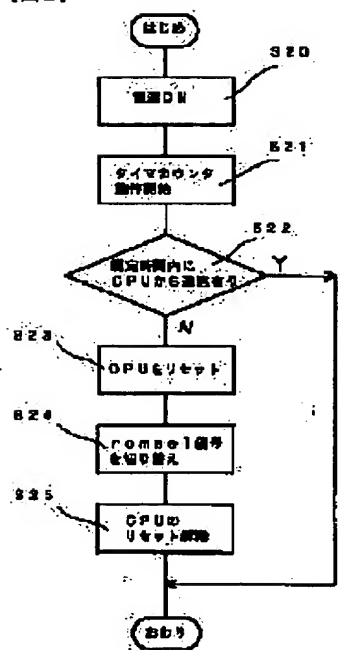
【符号の説明】

- 2 CPU
- 3 ブート制御用CPU
- 4 RS232Cコントローラ
- 5 アドレスデコーダ
- 6 第1のブートROM
- 7 第2のブートROM
- 11 ブートROM

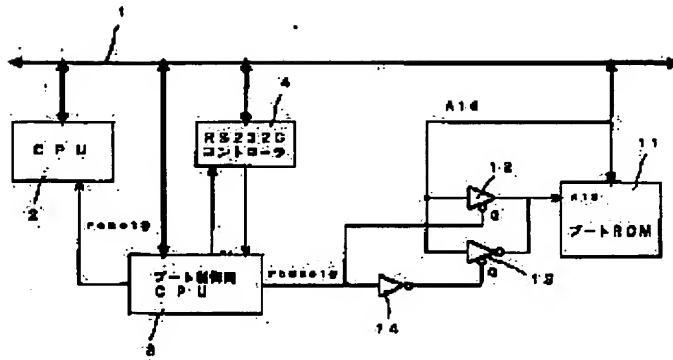
【図1】



【図2】



【図3】





**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☒ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**